JTC1/SC22/WG14 - N3123
Title: Clarification of Declaration Contexts
Author: Martin Uecker
Date: 2023-04-22

To address the issues raised by N3070 and to make the standard more readable, we suggest to make the following changes. Only Change 3 is intended to be a semantic change that removes unneeded restrictions for auto that the existing extensions do not have.

Change 1 (Clarify terminology)

6.7 Declarations
6 If a declaration is said to declare identifiers, this refers only to the identifiers that are declared by the declarators listed in its init-declarator-list. Nested declarations, type specifiers, or definitions of enumeration constants that are part of a declaration may declare additional identifiers.

Change 2 (Case 1 in N3070)

2 A declaration other than a static_assert or attribute declaration shall declare at least ~~a declarator (other than the parameters of a function or the members of a structure or union), a tag, or the members of an enumeration.~~ **one identifier, or specify at least one structure, union, or enumerated type using a type specifier that declares a tag, or specify at least one enumerated type that contains the definition for at least one enumeration constant.**

Change 3 (Case 2 in N3070)

5 ~~In an~~ **Identifiers declared by an** underspecified declaration ~~all declared identifiers that do not have a prior declaration~~ shall be ordinary identifiers. **xxx)**

xxx) Other identifiers may be declared as part of the initializer.

Change 4 (Case 3 in N3070)

6.7.6.2 Array declarators

4 If the size is not present, the array type is an incomplete type. If the size is * instead of being an expression, the array type is a variable length array type of unspecified size, which can only be used ~~in declarations or type names with function prototype scope~~ **as part of a parameter declaration.** 174)
174) ~~Thus,~~ **They** can be used only in function declarations that are not definitions (see 6.7.6.3 **and 6.9.1**)

6.2.1 Scopes of identifiers, type names, and compound literals

8 ~~As a special case, a type name (which is not a declaration of an identifier) is considered to have a scope that begins just after the place within the type name where the omitted identifier would appear were it not omitted.~~ A compound ...

6.9.1 Constraints:

X. Variable length array types of unspecified size shall not be used as part of a parameter declaration in a function definition.